WHAT IS CLAIMED IS:

1.   A memory storage structure, comprising:

at least one memory storage device;

a first meta-structure having a first size and

5   operating at a first speed, which is faster than a second

speed for storing meta-information based on information

stored in a memory;

a second meta-structure hierarchically associated with

the first meta-structures, the second meta-structure having

10   a second size larger than the first size and operating at

the second speed such that faster and more accurate

prefetching is provided by coaction of the first and second

meta-structures.

15   2.   The structure as recited in claim 1, wherein the

first and second meta-structures include branch history

tables and the meta-information includes branch history

data.

20   3.   The structure as recited in claim 2, further

-50-

comprising a predicted branch table for identifying a

sequence of predicted taken branches that a processor will

soon encounter.

5   4.   The structure as recited in claim 1, wherein the

meta-information includes temporally sequential information

that is likely to be used in the near future.

           5.   The structure as recited in claim 1, wherein the

10   meta-information includes spatially sequential information

that is likely to be used in the near future.

           6.   The structure as recited in claim 1, wherein the

meta-information is correlated to program flow in a

15   processor.

           7.   The structure as recited in claim 1, wherein the

at least one memory storage device includes a cache.

20   8.   The structure as recited in claim 7, wherein at

least one of the meta-structures are incorporated in the
cache.

9.   The structure as recited in claim 7, wherein the
cache is hierarchically arranged.

10.   The structure as recited in claim 9, wherein the
hierarchically arranged cache includes a first level cache
line and a second level cache line.

11.   The structure as recited in claim 1, further
comprising a meta-collector, which collects temporally
sequential unique meta-information entries corresponding to
a cache line.

12.   The structure as recited in claim 11, wherein the
unique meta-information includes at least one of a branch
address (BA) and a predicted target address (TA) for
information to be prefetched.

13. A memory storage structure, comprising:

a cache;

a meta-structure hierarchically arranged in accordance

with a size and speed such that faster and more accurate

5   prefetching is provided by coaction of hierarchical meta-

structures; and

a meta-collector which collects temporally and

spatially sequentially unique meta-information entries

corresponding to a cache line to enable the hierarchical

10   meta-structure operation.

14. The structure as recited in claim 13, wherein the

meta-structures include branch history tables and the meta-

information includes branch history data.

15

15. The structure as recited in claim 14, further

comprising a predicted branch table for identifying a

sequence of predicted taken branches that a processor will

soon encounter.

20

16. The structure as recited in claim 13, wherein the
meta-information is correlated to program flow in a
processor.

5   17. The structure as recited in claim 13, wherein at
least one meta-structure is incorporated in the cache.

18. The structure as recited in claim 13, wherein the
cache is hierarchically arranged.

10

19. The structure as recited in claim 18, wherein the
hierarchically arranged cache includes a first level cache
line and a second level cache line.

15   20. The structure as recited in claim 19, wherein the
meta-information includes at least one of a branch address
(BA) and a predicted target address (TA) for information to
be prefetched.

20   21. The structure as recited in claim 13, further

comprising a plurality of memory storage structures arranged

to prefetch information for stages of a circuit.

22. A method prefetching meta-information, comprising

5   the steps of:

providing a memory storage structure having a cache,

meta-structures hierarchically arranged in accordance with

size and speed, and a meta-collector which collects one of

temporally and spatially sequentially unique meta-

10   information entries corresponding to a cache line; and

prefetching meta-information for storage in the meta-

structures such that improved speed is provided by coaction

of hierarchical meta-structures.

15   23. The method as recited in claim 22, wherein the

step of prefetching includes associating cache lines with

information addresses in the meta-collector.

24. The method as recited in claim 22, wherein the

20   meta-structures include branch history tables and the meta-

information includes branch history data.

25.   The method as recited in claim 22, further comprising the step of identifying a sequence of predicted taken branches that a processor will soon encounter by employing a predicted branch table.

26.   The method as recited in claim 22, further comprising the step of correlating the meta-information to program flow in a processor.

27.   The method as recited in claim 22, further comprising the step of evicting cache line information from the meta-collector when a corresponding cache line is replaced.

28.   The method as recited in claim 22, further comprising the step of storing evicted information to a next level memory area in a cache hierarchy.

29.   The method as recited in claim 22, further comprising the step of on a cache miss, writing to a next level memory area in a cache hierarchy a cache miss address.

5   30.   The method as recited in claim 22, further comprising the step of updating meta-information between levels of the hierarchical meta-structures.

31.   The method as recited in claim 30, wherein the

10   step of updating includes updating meta-information by copying the meta-information between a level of a cache and a level of a branch history table.

32.   The method as recited in claim 30, wherein the

15   step of updating includes updating meta-information by copying the meta-information between a first level of a cache and a second level of a cache.

33.   The method as recited in claim 30, wherein the

20   step of updating includes updating meta-information by

copying the meta-information between a first level of a

meta-structure and a second level of a meta-structure.

34.   The method as recited in claim 30, wherein the

step of updating includes updating meta-information by

copying the meta-information to/from a meta-collector.

35.   The method as recited in claim 22, further

comprising the step of accumulating meta-information by

transferring the meta-information between entities such that

new meta-information is added upon each transfer.

36. A method for processing a temporal sequence of

events, wherein the events have spatial context, the method

comprising the steps of:

capturing a set of addresses in temporal order, the

addresses including information associated with each

address;

storing sub-sequences of temporal addresses which share

spatial context as monolithic entities wherein each

monolithic entity is associated with a particular spatial

context; and

when a new spatial context is encountered in the

temporal sequence, creating a new monolithic entity

5   associated with the new spatial context, the new spatial

context including the temporal sub-sequence of events

associated with the new spatial context.

37.   The methods as recited in claim 36, further

10   comprising the step of storing the monolithic entities

associated with the spatial contexts in their temporal order

of occurrence.

38.   The method as recited in claim 36, wherein the

15   monolithic entities include multi-dimensional data.

39.   The method as recited in claim 38 wherein one of

the multi-dimensions includes a spatial dimension.

20   40.   The method as recited in claim 38 wherein one of

the multi-dimensions includes a temporal dimension.

41.   The method as recited in claim 38 wherein one of the multi-dimensions includes metadata.

5

42.   The method as recited in claim 36 wherein the information includes metadata.

43.   The method as recited in claim 36, further

10   comprising: storing the monolithic entities at a location determined by spatial context of the monolithic entities.

44.   The method as recited in claim 36, further comprising: storing the monolithic entities at a location

15   determined by temporal context of the monolithic entities.

45.   The method as recited in claim 36, further comprising: retrieving monolithic entities from storage in accordance with spatial content of the said monolithic

20   entities.

46.   The method as recited in claim 45, further comprising: using metadata associated with the monolithic entities by a processor after the monolithic entities are

5   retrieved.

47.   The method as recited in claim 36, further comprising: retrieving monolithic entities from storage in accordance with temporal content of the said monolithic

10   entities.

48.   The method as recited in claim 47, further comprising: using metadata associated with the monolithic entities by a processor after the monolithic entities are

15   retrieved.